A Survey of Some Methods for Real Quantifier Elimination, Decision, and Satisfiability and Their Applications

Thomas Sturm

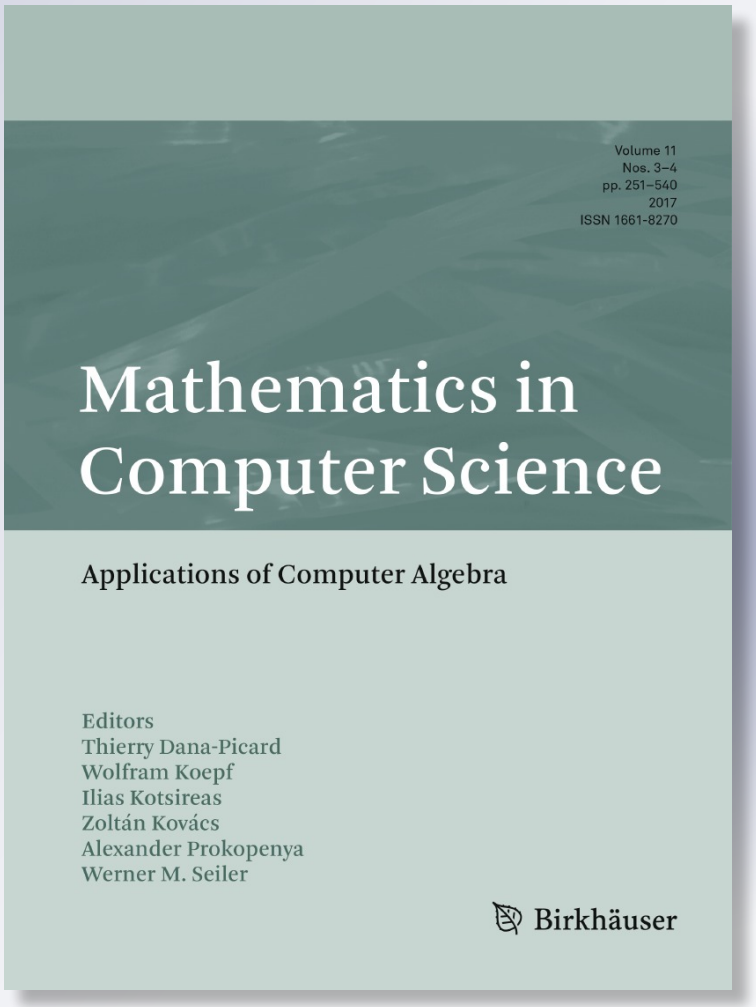
A Survey of Some Methods for Real Quantifier Elimination, Decision, and Satisfiability and Their Applications

Thomas Sturm

Mathematics in Computer Science

ISSN 1661-8270
Volume 11
Combined 3-4

Math.Comput.Sci. (2017) 11:483-502
DOI 10.1007/s11786-017-0319-z



Springer

Math.Comput.Sci. (2017) 11:483–502
DOI 10.1007/s11786-017-0319-z

Mathematics in Computer Science

A Survey of Some Methods for Real Quantifier Elimination, Decision, and Satisfiability and Their Applications

Thomas Sturm

Received: 18 December 2016 / Revised: 20 March 2017 / Accepted: 21 March 2017 / Published online: 26 April 2017

Abstract Effective quantifier elimination procedures for first-order theories provide a powerful tool for generically solving a wide range of problems based on logical specifications. In contrast to general first-order provers, quantifier elimination procedures are based on a fixed set of admissible logical symbols with an implicitly fixed semantics. This admits the use of sub-algorithms from symbolic computation. We are going to focus on quantifier elimination for the reals and its applications giving examples from geometry, verification, and the life sciences. Beyond quantifier elimination we are going to discuss recent results with a subtropical procedure for an existential fragment of the reals. This incomplete decision procedure has been successfully applied to the analysis of reaction systems in chemistry and in the life sciences.

Keywords Real quantifier elimination and decision · Satisfiability · Virtual substitution · Subtropical methods · Real geometry · Verification · Reaction systems · Stability analysis

Mathematics Subject Classification 68U99

1 Introduction

This survey article is based on various invited conference talks, each with a different focus, which I gave at SMT 2013 in Helsinki, Finland, at FroCos 2015 in Wrocław, Poland, and at ACA 2016 in Kassel, Germany. It finally consolidates the existing material and will, hopefully, provide an interesting reference for the application of real algebra, symbolic computation, and related logical methods in mathematics, engineering, and the sciences. Figure 1 gives an impression on topics and research areas relevant for the material presented here.

The plan of this article is as follows: In Sect. 2 we introduce and discuss real quantifier elimination and variants with a strong focus on virtual substitution methods. For cylindrical algebraic decomposition, which is another practically applicable method with strong implementations we refer the reader to an introduction by Jirstrand [32]

T. Sturm (✉)
University of Lorraine, CNRS, Inria, and LORIA, Nancy, France
e-mail: thomas@thomas-sturm.de
URL: http://science.thomas-sturm.de

T. Sturm
Max Planck Institute for Informatics and Saarland University, Saarbrücken Informatics Campus, Saarbrücken, Germany

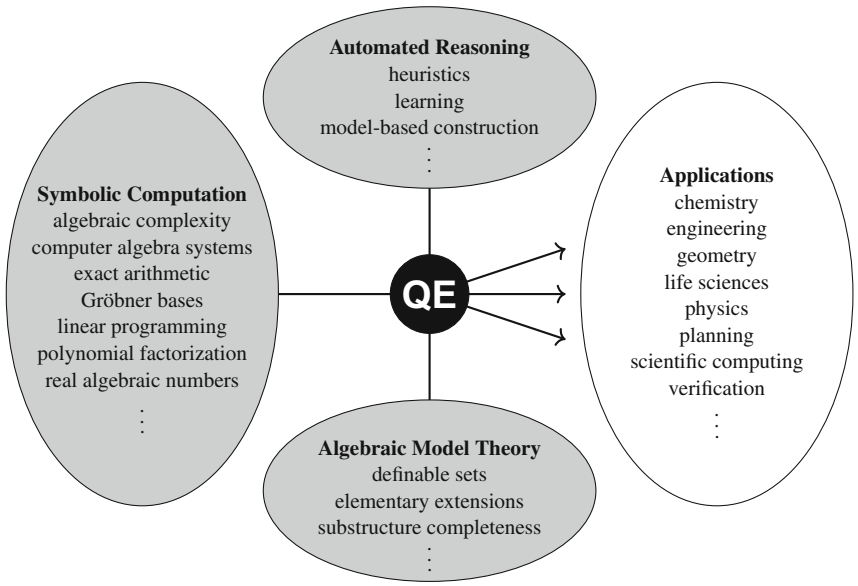


Fig. 1 Quantifier elimination-relevant research topics

or to tutorial material by Brown used at ISSAC 2004,[1] both of which are excellent. In Sect. 3 we discuss the application of real quantifier elimination and variants thereof to geometric theorem proving, verification, and bifurcation analysis in the life sciences. In Sect. 4 we introduce an incomplete heuristic satisfiability checking procedure for the reals in order to consider in Sect. 5 larger examples from the life sciences and also from chemistry. The applications here are very similar to the life science applications discussed in detail in Sect. 3. Therefore[2] we only summarize the models and focus on the current limit of problem sizes tractable by symbolic methods in that area. We end with some concluding remarks in Sect. 6.

This article is a survey of my own work, partly with collaborators. There is no claim that it contains unpublished original results. I allowed myself to take over single sentences or short passages from my own publications without explicitly mentioning this.

2 Real Quantifier Elimination and Variants

2.1 Introduction

The following formal statement φ over the reals asks whether or not one can find for all $x \in \mathbb{R}$ some $y \in \mathbb{R}$ such that a certain polynomial $p \in \mathbb{Z}[a, b, x, y]$ is strictly positive while another such polynomial q is zero or negative:[3]

$$\varphi \doteq \forall x \exists y (p > 0 \wedge q \leq 0) \quad \text{where} \quad p \doteq x^2 + xy + b, \quad q \doteq x + ay^2 + b.$$

We have to expect that the validity of φ depends on the choices of real values for the *parameters* a and b. Even for readers familiar with real algebraic geometry, a solution is probably not easy to see.[4] It gets easier when considering

[1] https://www.usna.edu/Users/cs/wcbrown/research/ISSAC04/Tutorial.html.

[2] And also due to reasonable page limits imposed by the editors.

[3] This instructive example has been used by Hoon Hong in a colloquium talk at the University of Passau in 2005.

[4] The free tool GeoGebra greatly supports intuition by plotting the ranges of the two inequalities with respect to $(x, y) \in \mathbb{R}^2$ and allowing transformations of that plot via sliders for the values of a and b; our example can be found at https://www.geogebra.org/m/f7UG3zpq.

$\neg\varphi$, which is equivalent to $\exists x \forall y (p \leq 0 \vee q > 0)$. When $a \geq 0$, we can choose $x = -b + 1$ to satisfy $q > 0$. When $b \leq 0$, we can choose $x = 0$ to satisfy $p \leq 0$. Thus $\bar{\varphi} \doteq a \geq 0 \vee b \leq 0$ implies $\neg\varphi$. Equivalently, φ implies $\varphi' \doteq a < 0 \wedge b > 0$. Vice versa, it is not hard to see that φ' implies φ: Given $x \in \mathbb{R}$ choose $y \in \mathbb{R}$ with $\operatorname{sgn}(y) = \operatorname{sgn}(x)$ and $|y| = \sqrt{|(x+b)/a|}$.

Formally, we are considering interpreted first-order logic with equality over a finite language[5] $L = (0, 1, +, -, \cdot, \leq)$ where all symbols have their usual interpretations over the reals. We admit ourselves to use relations "$<$", "$>$", "$\geq$", "$\neq$" and integers as short notations for corresponding quantifier-free formulas and terms, respectively. Given a first-order L-formula φ, our goal is to compute a quantifier-free L-formula φ' such that $\mathbb{R} \models \underline{\forall}(\varphi \longleftrightarrow \varphi')$, where "$\underline{\forall}$" denotes the universal closure. This is called real *quantifier elimination*. We call φ an *L-sentence*, if all occurring variables are quantified, in other words, if there are no parameters. When applying quantifier-elimination to a sentence φ, the obtained quantifier-free formula φ' will not contain any variables and can be straightforwardly simplified to either "true" or "false". This way, real quantifier elimination establishes in particular a *decision procedure*. In the special case that φ is of the form $\exists\psi$, where "$\underline{\exists}$" denotes the existential closure and ψ is quantifier-free, quantifier elimination establishes a *satisfiability checking procedure*. In Sect. 4 we will study a satisfiability checking approach that is not based on quantifier elimination.

In symbolic computation our formal framework is often called the *Tarski Algebra*, while in satisfiability modulo theories solving (SMT) it is referred to as *nonlinear real arithmetic*. One might mention that from a point of view of algebraic model theory we are considering the model class $\mathrm{RCF} = \mathrm{Mod}(\mathrm{Th}(\mathbb{R}))$ of real closed fields, containing all L-structures in which exactly the same first-order L-sentences hold as in $\mathbb{R}$. Natural enumerable axiomatizations of RCF can be found in [46]. By the Löwenheim–Skolem Theorem, RCF is a proper class containing models of arbitrary infinite cardinality. Among those models are the real numbers $\mathbb{R}$, the countable model of all real algebraic numbers, or the non-Archimedian extension field $\mathbb{R}(\infty)$, in which ∞ is algebraically free over $\mathbb{R}$ but strictly greater than all real numbers.[6] Since from the point of view of first-order L-formulas, the models in RCF cannot be distinguished, it is safe to assume with the application of real quantifier elimination that the domain of the computation is $\mathbb{R}$, or any other model in RCF. We shall see that under the hood of real quantifier elimination and decision procedures, non-standard models, in particular the ones mentioned above, play an important role.

2.2 History

The first real quantifier elimination procedure was published by Tarski at the end of the 1940s [65].[7] As early as 1954, concluding remarks in a technical report by Davis to the US Army on an implementation of a similar procedure for Presburger arithmetic point at a very early interest in software implementations of real quantifier elimination [16].

During the 1970s Collins developed the first elementary recursive real quantifier elimination procedure [11,12], which was based on *cylindrical algebraic decomposition* (CAD). An implementation by Arnon was available around 1980 [1]. CAD has undergone many improvements since and establishes an active research area until today [7,14,40]. Hong reimplemented CAD in an interactive software Qepcad [13]. Qepcad developed into Qepcad B, which is now maintained by Brown [6].

From the mid 1980s to the early 1990s there was a strong interest in the asymptotic worst-case time complexity of the decision problem for RCF. In 1988 Davenport–Heintz [15] and Weispfenning [72] independently showed that it is doubly exponential. Weispfenning's article actually brought even stronger results: first, it showed that the decision problem is doubly exponential already for linear formulas, where the total degree in all quantified variables does not exceed 1. Next, considering finer complexity parameters than the input word length it showed

[5] Alternatively called a *signature.*

[6] Certainly all these models are unique only up to isomorphisms.

[7] Tarski's results are actually around 10 years older, but publication was delayed due to World War II.

that the problem for linear formulas is doubly exponential in the number of quantifier alternations but only singly exponential in the number of quantifiers (when bounding the number of alternations) and only polynomial in the number of variables (when bounding the number of quantifiers). Finally, it came with a corresponding effective procedure called *virtual substitution* (VS), where the idea is for the elimination of an existential quantifier to formally substitute in a generalized sense sufficiently many parametric zeros of certain polynomials contained in the input formula. Weispfenning's complexity results were followed by further research by Grigoriev, Renegar, Basu–Pollack–Roy, and others developing entirely new quantifier elimination procedures with strong theoretical results taking into consideration even finer complexity parameters like polynomial degrees or coefficient sizes [2,27,48].[8] Those asymptotically fast procedures remained unimplemented, and there are in fact substantial arguments that the trade-off points are too large for the procedures to be relevant in practice [30].

VS, in contrast, turned out to be practical. It was implemented by Weispfenning's students in the computer logic system Redlog [17,19,61], which has developed since into an established tool with more than 350 citations in the scientific literature for successful applications mostly in the sciences.[9] The focus is on low-degree problems with few quantifier alternations and with a significant number of parameters, as we know today that CAD—and thus Qepcad B—is doubly exponential in *all* variables. All examples discussed in this article have been computed with an implementation of VS in Redlog that is limited to formulas where for the elimination of a quantifier the total degree of the corresponding quantified variable does not exceed 2. Nevertheless we will encounter significantly higher degrees, because VS implementations are supplemented with powerful simplification strategies; see, e.g., [20] and [36, Section 5]. The nonlinear bound of 2 is based on work by Weispfenning in the early 1990s, which also theoretically discussed arbitrary degree bounds [73]. Only recently, Košta has theoretically improved Weispfenning's framework for higher degrees and provided a generic implementation in Redlog, which can be instantiated with substitution tables up to a chosen degree bound [35]. Highly optimized such tables are available up to degree 3; providing practically useful tables for higher degrees is a realistic but challenging task.

2.3 Degree Bounds

It is important to understand that the degree bounds for VS have to be satisfied for the quantified variable with the elimination of every single quantifier. Quantifiers are essentially eliminated one by one. Except for Weispfenning's original *linear case* [39,72], where there are no products of quantified variables at all, the elimination of a quantifier will in general increase the degrees of other quantified variables. As a consequence, one cannot tell by inspection of the input whether or not quantifier elimination will succeed subject to the current degree bound. This in turn makes it hard to give meaningful upper complexity bounds. In fact, Weispfenning's bounds discussed above have been proved only the linear case. From a practical point of view, hundreds of meaningful problems have been solved during the past 25 years with a degree bound of only 2. Those computations have given the following impression: Firstly, input formulas modelling real world situations, in contrast to randomly generated ones, are surprisingly well-behaved concerning the increase of degrees during elimination. Secondly, the observed computation times appear compatible with the theoretical results for the linear case concerning improved performance with bounded quantifier alternation and significantly less sensitivity to numbers of parameters than to numbers of quantified variables.

2.4 Virtual Substitution in a Nutshell

Every L-formula can be equivalently transformed into a *prenex* L-formula consisting of a sequence of quantifiers followed by a quantifier-free formula. Without loss of generality, all right hand sides of equations and inequalities

[8] The lower bounds by Davenport–Heintz [15] and Weispfenning [72] as well as Grigoriev's [27] upper bound all appeared in the same issue of the Journal of Symbolic Computation in 1988. Weispfenning's article had been submitted already in 1983.

[9] Based on an analysis of Google citation counts for the standard reference [19] for Redlog.

are 0. Given such a prenex formula, VS successively eliminates the quantifiers starting with the innermost one. Using the equivalence $\forall x_1 \psi \longleftrightarrow \neg\exists x_1 \neg\psi$ we may w.l.o.g. assume that the innermost quantifier is an existential one. More generally, there is an innermost block of finitely many existential quantifiers to be eliminated, possibly with further quantifier blocks outside. The idea is to replace $\exists x_1$ with a finite disjunction such that

$$\exists x_n \ldots \exists x_2 \exists x_1(\psi) \longleftrightarrow \exists x_n \ldots \exists x_2 \bigvee_{(\gamma,t)\in E} \gamma \wedge \psi[x_1 /\!\!/ t],$$

where $E = \{\ldots, (\gamma, t), \ldots\}$ is a finite *elimination set* containing sufficiently many test points t. Notice that before the next elimination step $\exists x_2$ can be moved into the disjunction. This explains the better performance of the procedure with bounded quantifier alternation.

Fixing an interpretation of all variables except x_1 it is not hard to see that the set of satisfying values for ψ with respect to x_1 is a finite union of intervals. Furthermore, the upper and lower bounds of those intervals are $-\infty$, ∞, or zeros of the left hand side polynomials in ψ. On these grounds, the test points t in E are derived as formal solutions of the polynomials in ψ, and the corresponding *guards* γ guarantee the existence of those solutions. For instance, elimination set elements produced by $ax^2 - 3x + 7 \leq 0$ would include

$$\left(a \neq 0 \wedge (-3)^2 - 4 \cdot a \cdot 7 \geq 0,\ \frac{3-\sqrt{(-3)^2-4\cdot a\cdot 7}}{2\cdot a}\right), \quad \left(a = 0,\ \tfrac{7}{3}\right).$$

However, standard term substitution is not sufficient here: For instance, we must substitute quotients and square roots, both of which are not in our language L. Furthermore, with strict inequalities we need, e.g., $t - \varepsilon$ with infinitesimal ε, and similarly we need ∞. Those substitutions must be carried out in such a way that the substitution results contain only symbols from L. The key idea is to use a *virtual substitution*, which does not map terms to terms but atomic L-formulas to quantifier free L-formulas. We give some concrete examples, where $f \in \mathbb{Z}[\mathbf{y}][x]$ and $f_i, g_i, g_i^* \in \mathbb{Z}[\mathbf{y}]$:

Quotients:

$$(f_1 x + f_0 \leq 0)\left[x /\!\!/ \tfrac{g_1}{g_2}\right] \doteq f_1 \tfrac{g_1}{g_2} + f_0 \leq 0 \doteq f_1 g_1 g_2 + f_0 g_2^2 \leq 0.$$

Infinity:

$$(f_2 x^2 + f_1 x + f_0 < 0)[x /\!\!/ \infty] \doteq f_2 < 0 \vee (f_2 = 0 \wedge f_1 < 0) \vee (f_2 = 0 \wedge f_1 = 0 \wedge f_0 < 0).$$

Positive infinitesimals:

$$(3x^2 + 6x - 3 > 0)[x /\!\!/ t - \varepsilon] \doteq 3t^2 + 6t - 3 > 0 \vee (3t^2 + 6t - 3 = 0 \wedge 6t + 6 \leq 0).$$

Formal solutions of quadratic equations:

$$(f = 0)\left[x /\!\!/ \tfrac{g_1 + g_2\sqrt{g_3}}{g_4}\right] \doteq \tfrac{g_1^* + g_2^*\sqrt{g_3}}{g_4^*} = 0 \doteq g_1^{*2} - g_2^{*2} g_3 = 0 \wedge g_1^* g_2^* \leq 0.$$

It turns out that virtual substitution is actually strong enough to substitute abstract formal representations of roots of polynomials of arbitrary degrees with suitable guards. This explains the role of the substitution tables with the new generic implementation mentioned at the end of Sect. 2.2 [35].

2.5 Variants of Real Quantifier Elimination

There are numerous variants and extensions of the concept of quantifier elimination. We are going to discuss here and use with our applications in the following section three of them.

2.5.1 Extended Quantifier Elimination

Extended quantifier elimination [73] does not form a disjunction over the virtual substitution results. Instead, they are kept separate and associated with formal assignments of the substituted test points:

$$\exists x_1 \psi \;\rightsquigarrow\; \begin{bmatrix} \vdots & \vdots \\ \gamma \wedge \psi[t/\!\!/x_1] & x_1 = t \\ \vdots & \vdots \end{bmatrix}.$$

There is a precise semantics as follows: *For fixed choices of all parameters, whenever some left hand side condition holds, then* $\exists x_1 \psi$ *holds, and the corresponding right hand side assignment is* one *sample solution.* As a simple example consider

$$\exists x(ax^2 + bx + c = 0) \;\rightsquigarrow\; \begin{bmatrix} a \neq 0 \wedge b^2 - 4ac \geq 0 & x = \frac{-b-\sqrt{b^2-4ac}}{2a} \\ a \neq 0 \wedge b^2 - 4ac \geq 0 & x = \frac{-b+\sqrt{b^2-4ac}}{2a} \\ a = 0 \wedge b \neq 0 & x = -\frac{c}{b} \\ a = 0 \wedge b = 0 \wedge c = 0 & x = \infty_1 \end{bmatrix}.$$

Notice that our specification of the semantics above would allow the simplification component of virtual substitution to delete one of the first two rows from the result. In the last row we can see that an infinite element has been introduced as a test point. In the course of the elimination of further variables this can happen several times, which corresponds to a stack of extensions of real closed fields, where every newly introduced infinite element is infinite not only compared to standard real numbers but also compared to the already existing infinite elements. Therefore all such elements are carefully distinguished using indices.

With the elimination of several existential quantifiers one obtains assignments for all corresponding variables, on which one can perform back-substitution in the style of linear programming. For fixed choices of parameters, non-standard elements can be efficiently eliminated from the sample solutions in a post-processing step [36]. Finally, with universal quantifiers we have the following dual semantics: *For fixed choices of all parameters, whenever some left hand side condition does not hold, then* $\forall x_1 \psi$ *does not hold, and the corresponding right hand side assignment is* one *counterexample.*

2.5.2 Generic Quantifier Elimination

The combinatorial complexity of virtual substitution mostly arises from case distinctions on the vanishing of certain coefficients. We have seen an example with the test points derived from $ax^2 - 3x + 7 \leq 0$ in Sect. 2.4. The idea of generic quantifier elimination [18,54,62] is to avoid those case distinctions when the corresponding coefficient does not contain any quantified variables. It assumes the generic case that the coefficient does not vanish. Those assumptions are collected in a global *theory* Θ, which establishes part of the output:

$$E = \{\ldots, (s \neq 0 \wedge \gamma', t), \ldots) \;\rightsquigarrow\; \Theta = \{\ldots, s \neq 0, \ldots\},\; E = \{\ldots, (\gamma', t), \ldots\}.$$

Again, there is a precise semantics stating that the elimination result obtained this way is correct for all choices of parameters that satisfy Θ, formally:

$$\mathbb{R} \models \underline{\forall}\big(\textstyle\bigwedge \Theta \longrightarrow (\varphi \longleftrightarrow \varphi')\big).$$

Consider again our simple example from the previous section:

$$\varphi \doteq \exists x(ax^2 + bx + c = 0) \;\rightsquigarrow\; \Theta = \{a \neq 0\}, \quad \varphi' \doteq b^2 - 4ac \geq 0.$$

Notice that we only assume negations of equations but no ordering inequalities. Hence Θ cannot become inconsistent. Furthermore, the set of choices for the parameters that does not satisfy the assumptions in Θ has a lower dimension than the parameter space.

2.5.3 Positive Real Quantifier Elimination

With many natural problems, variables may be assumed to be positive. The idea of positive quantifier elimination is to have a variant of the elimination procedure which systematically exploits that assumption [57,64]. First implementations of that idea, as the one used with our life science application in Sect. 3.3, implicitly assumed that all occurring variables are strictly positive. With a more modern approach now taken in Redlog relevant positivity conditions are explicitly encoded in the input formula and extracted by the VS procedure. This way positive quantifier elimination turns into an optimization of the procedure which is transparent for the user.

3 Applications of Real Quantifier Elimination

3.1 Geometry

We focus here on geometric theorem proving. Related applications of real quantifier elimination methods include computational geometry [58] and solid modeling [59,60]. Theorems of elementary geometry have traditionally been considered an important test case for the scope of methods in automatic theorem proving. In particular, they have stimulated a variety of algebraic techniques for their solution. There is an established style of algebraic modeling with parameters $\mathbf{u}$ and dependent variables $\mathbf{x}$, which can be found already in Hilbert's *Grundlagen der Geometrie*[10] [29], originally published in 1899. Using a suitably positioned coordinate system, a geometric configuration and a claimed geometric property of that configuration are translated into a formula

$$\varphi \doteq \bigwedge_i h_i(\mathbf{u}, \mathbf{x}) = 0 \longrightarrow g(\mathbf{u}, \mathbf{x}) = 0,$$

where polynomial equations with $h_i \in \mathbb{Z}[\mathbf{u}, \mathbf{x}]$ and $g \in \mathbb{Z}[\mathbf{u}, \mathbf{x}]$ describe the configuration and the claim, respectively. The parameters $\mathbf{u}$ describe geometric entities that can be freely chosen, e.g., three corners of a triangle, while the dependent variables $\mathbf{x}$ correspond to entities that are uniquely determined by the existing construction, e.g., the center of the circumcircle of that triangle.

Around 1977, Wu discovered the *Wu–Ritt method*[11] [74,75][12] adapting a method for differential algebra developed by Ritt [49,50]. Chou made important contributions to the development and application of the Wu–Ritt method. Most of Chou's work has been summarized in an impressive monograph including more than 500 geometric theorems automatically proved by this method [9]. Kapur [34] developed a prover based on a radical membership test via Gröbner Basis [8] computations. Kutzler and Stifter [37] developed another prover also based on Gröbner techniques. They modified the notion of reduction to a *pseudo reduction*, which is somewhat similar to the notion of reduction used with the Wu–Ritt method. Wang [69,70] developed complex elimination methods based on ideas by Seidenberg [51–53] but also inspired by Wu–Ritt.

All those methods, which have turned out to be quite successful, try to prove φ as a statement about *complex numbers*. Since φ is a universal assertion, the validity of φ over the complex numbers entails the validity of φ over the reals and thus an automatic proof of the original geometric assertion. If, in contrast, φ turns out to be false over the complex numbers, no decision on the validity of the geometric statement can be made. It is due to the historical focus on complex methods that the algebraic translations introduced above are purely equational. From a geometric point of view it makes sense to also have inequalities comparing, e.g., lengths. Employing real quantifier elimination will allow us to do so.

Before studying an example in detail, we want to have a closer look at the available freedom with the parameters $\mathbf{u}$. We had mentioned earlier than one would, e.g., introduce (u_1, u_2), (u_3, u_4), (u_5, u_6) as corners of a triangle in the

[10] Foundations of Geometry.

[11] Also known as *Wu–Ritt reduction* or *Wu's method.*

[12] This article appeared in two different journals in 1984 and 1986. The reason was limited availability of the first one. Both these publications are in English. We thank Xiao-Shan Gao for clarifying this.

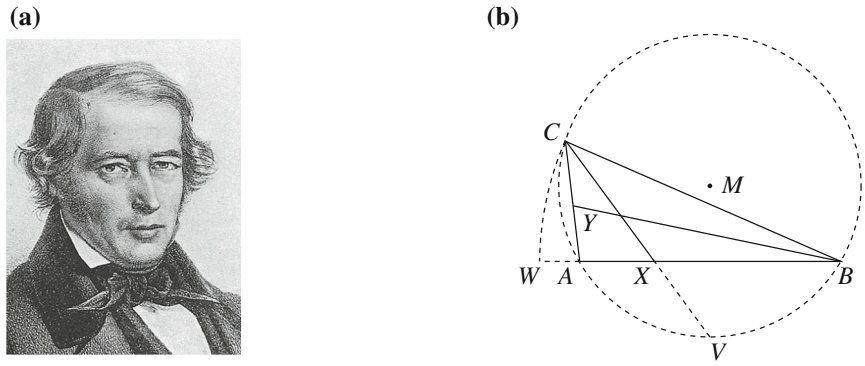


Fig. 2 The Steiner–Lehmus Theorem. **a** The Swiss mathematician Jakob Steiner.[13] **b** Construction for our variant of the theorem

plane. However, this requires certain *non-degeneracy conditions* stating, e.g., that those three points be not collinear. When they are, they do not form a triangle. This breaks the geometric construction and in general also the validity of φ. Furthermore, for a geometric theorem to hold it might be necessary to exclude certain valid configurations like parallel lines or rectangular triangles. Since such cases are not really degenerate, non-degeneracy conditions have also been called *subsidiary conditions*. It is in general by no means easy to find and add all necessary subsidiary conditions for the validity of a theorem. Illustrating examples for hidden conditions have been given by Chou [9, pp. 43–44].

As a matter of fact, the results produced with the above-mentioned algebraic methods are correct also only under certain assumptions in the shape of negated equations. Those assumptions would state, e.g., that during the algebraic computations certain parametric polynomial coefficients $c(\mathbf{u})$ do not vanish. They are collected and explicitly provided with the output. We are going to apply here the generic real quantifier elimination from Sect. 2.5.2, which also produces such assumptions. It is a most amazing fact that the assumptions made by the proof methods mostly coincide with the subsidiary conditions necessary for the validity of the corresponding geometric theorem [9,18,62]. That is, an apparent weakness of the algebraic procedures turns out as an adequate tool for actively *discovering* missing hypotheses. The presence of this phenomenon with many different algebraic methods suggests that the approach to translate geometry to algebra is very natural.

We are going to study a variant of the *Steiner–Lehmus Theorem* taken from McPhee et al. [41]. In its original form the theorem states that *any triangle with two equal internal bisectors is isosceles.* It had been firstly mentioned in a letter by Christian Ludolf Lehmus to Charles-François Sturm in 1840, asking for a purely geometric proof. Sturm passed the question on to several mathematicians, and Jakob Steiner provided such a proof. The contrapositive of the original Steiner–Lehmus Theorem follows immediately from the following, stronger, variant; compare Fig. 2b:

> **Steiner–Lehmus Theorem** (**variant**). *Assume that ABC is a triangle such that $AB > AC$. Then the angle bisector from B to AC is longer than the angle bisector from C to AB. That is, the longer bisector goes to the shorter side.*

For our algebraic translation, we take coordinates $A = (-1, 0)$, $B = (1, 0)$, and $C = (u_1, u_2)$ for the triangle. We assume w.l.o.g. that both C and the center $M = (0, x_1)$ of the circumcircle of ABC lie above AB:

$h_1 \doteq u_2 \geq 0 \wedge x_1 \geq 0.$

The center $M = (0, x_1)$ of the circumcircle is determined by its radius:

$h_2 \doteq r^2 = 1 + x_1^2 = u_1^2 + (u_2 - x_1)^2.$

We now construct the bisector of $\angle ACB$: The point $V = (0, x_2)$ is the lower extremity of the circumcircle:

$h_3 \doteq x_2 \leq 0 \wedge r^2 = (x_2 - x_1)^2,$

[13] This image, taken from Wikipedia, is in the public domain.

and $X = (x_3, 0)$ is the intersection between CV and AB. That is, the points CXV are collinear:

$$h_4 \doteq u_1 x_2 + u_2 x_3 - x_2 x_3 = 0.$$

The line segment CX is the bisector on the side AB. For constructing the bisector on AC using the above technique, we would have to guarantee that both M and B lie on the same side of AC within the circumcircle, which would introduce tedious case distinctions. We thus construct the bisector on AC more straightforwardly: Let $W = (x_4, 0)$ be the point on the line AB that lies left of B with distance BC:

$$h_5 \doteq x_4 \leq 1 \wedge (x_4 - 1)^2 = (u_1 - 1)^2 + u_2^2.$$

Then the foot $Y = (x_5, x_6)$ of the bisector on AC is the unique point with equal distance to both W and C and with AYC collinear:

$$h_6 \doteq (x_4 - x_5)^2 + x_6^2 = (u_1 - x_5)^2 + (u_2 - x_6)^2 \wedge u_1 x_6 - u_2 x_5 - u_2 + x_6 = 0.$$

Finally we add to our hypothesises the fact that AC is shorter than AB:

$$h_7 \doteq (-1 - u_1)^2 + u_2^2 < 2^2$$

and state the conclusion that CX is shorter than BY:

$$g \doteq (u_1 - x_3)^2 + u_2^2 < (x_5 - 1)^2 + x_6^2.$$

In terms of these definitions of $h_1, \ldots, h_7$ and g, our algebraic translation of the Steiner–Lehmus Theorem reads as follows:

$$\varphi \doteq \forall x_6 \forall x_5 \forall x_4 \forall x_3 \forall x_2 \forall x_1 \forall r \left(\bigwedge_{i=1}^{7} h_i \longrightarrow g \right).$$

Application of generic quantifier elimination to φ yields a quantifier-free formula φ' containing 231 atomic formulas plus the following assumptions:

$$\Theta = \left\{ u_1^2 - 2u_1 + u_2^2 - 3 \neq 0, u_1 \neq 0 \wedge u_2 \neq 0 \right\}.$$

The first assumption can be rewritten as $(u_1 - 1)^2 + u_2^2 \neq 4$, i.e., the distance between B and C is different from 2. In other words, the triangle ABC is not isosceles. The assumption $u_1 \neq 0$ also states that the triangle is not isosceles, and $u_2 \neq 0$ is a natural non-degeneracy condition for the triangle. While with other examples one frequently obtains $\varphi' \doteq \text{true}$, the situation here is more complicated. However, applying cylindrical algebraic decomposition to $\forall u_1 \forall u_2 \left(\bigwedge \Theta \longrightarrow \varphi' \right)$ we obtain "true", which means that φ' is indeed true under the assumptions already made. The overall computation time for this example is around 2 s.

3.2 Verification

Given a system, typically modeled by differential equations, and given a property, the verification problem asks whether or not the system satisfies the property. We are going to prove collision avoidance for two cars under cruise control [56]. The rear car uses a cruise control law that actively adjusts its acceleration based on its own velocity, acceleration, the relative velocity of the leading car, and the gap between the two cars. Proving collision avoidance means showing that the two cars will not collide assuming that the cruise control is activated in a safe initial configuration. The cruise control law is taken from the leader control developed in [26] and also discussed in [47,66]; see also the Berkeley Path project.[14] The system dynamics and the control law are given as a system of ordinary differential equations in Fig. 3, where $\dot{f} := \mathrm{d}\, f / \mathrm{d}\, t$ is the common short notation for the derivative with respect to time.

[14] http://www.path.berkeley.edu.

$$\begin{aligned}
\dot{v}_f &= a_f \in [-5, 2] && \text{velocity and accelleration of the leading car}\\
\dot{v} &= a \in [-5, 2] && \text{velocity and accelleration of the rear car}\\
\dot{\text{gap}} &= v_f - v && \text{gap between the two cars}\\
\dot{a} &= -3a - 3(v - v_f) + \text{gap} - (v + 10) && \text{control law for the rear car}
\end{aligned}$$

Fig. 3 A system of ordinary differential equations describing the system dynamics and the control law for our verification problem

For the initial state we choose a gap, assume that the rear car is driving at constant speed, and introduce parameters c_1 and c_2 for the speeds of the leading and the rear car, respectively:

$\text{Init} \doteq \text{gap} = 10 \wedge a = 0 \wedge v_f = c_1 \wedge v = c_2.$

A state is defined to be safe whenever there is a positive gap between the two cars. Of course, there is an additional safety distance, which is not made explicit in our model:

$\text{Safe} \doteq \text{gap} > 0.$

We use the *certificate-based approach* for verification [28,45,66], where the idea is to discover an *invariant set* Inv, which is defined by the following properties:[15]

(I_1) $\text{Init} \subseteq \text{Inv}$
(I_2) $\text{Inv} \subseteq \text{Safe}$
(I_3) The system dynamics does not admit the system to leave the set Inv.

Such a set Inv serves as a certificate for safety. We make an ansatz assuming Inv can be defined in terms of the non-negativity of a suitable linear combination of entities in our model. We may normalize one of the linear factors to 1:

$\text{Inv} \doteq p \geq 0, \quad \text{where} \quad p \doteq \lambda_1 v + \lambda_2 v_f + \lambda_3 a + \text{gap} + \lambda_4.$

The system dynamics can be over-approximated in Tarski algebra as follows:

$\text{Dyn} \doteq -5 \leq a \wedge a \leq 2 \wedge -5 \leq a_f \wedge a_f \leq 2 \wedge v \geq 0 \wedge v_f \geq 0.$

In terms of these definitions we formally write down the certificate conditions (I_1)–(I_3):

$\varphi_1 \doteq \text{Dyn} \wedge \text{Init} \longrightarrow \text{Inv}$
$\varphi_2 \doteq \text{Dyn} \wedge \text{Inv} \longrightarrow \text{Safe}$
$\varphi_3 \doteq \text{Dyn} \wedge p = 0 \longrightarrow \dot{p} \geq 0.$

The derivative of p in φ_3 can be symbolically computed using the system dynamics, which specifies that accelerations are derivatives of velocities, and the control law, which explicitly yields the derivative of the acceleration a of the rear car:

$$\dot{p} \doteq \lambda_1 a + \lambda_2 a_f + \lambda_3\big(-3a - 3(v - v_f) + \text{gap} - (v + 10)\big) + (v_f - v).$$

On the basis of these preparations the following formula with parameters c_1 and c_2 corresponding to initial speeds of the two cars asks for the existence of linear factors $\lambda_1, \ldots, \lambda_4$ such that Inv is in fact an invariant:

$$\varphi \doteq \exists\lambda_1\exists\lambda_2\exists\lambda_3\exists\lambda_4\forall v\forall v_f\forall\text{gap}\forall a\forall a_f(\varphi_1 \wedge \varphi_2 \wedge \varphi_3).$$

Redlog's virtual substitution eliminates all variables except λ_3, where it finally fails due to the degree bound.[16] The result is a disjunction φ'_{584} of 584 subformulas altogether containing 33,365 atomic formulas in our parameters

[15] Notice that there is natural correspondence between our formulas and their respective sets of satisfying values in a real space of suitable dimension. For the sake of a concise description we do not make this explicit.

[16] Recall that with the elimination of a block of either existential or universal quantifiers the order of the variables can be freely chosen, and Redlog uses heuristics to do so.

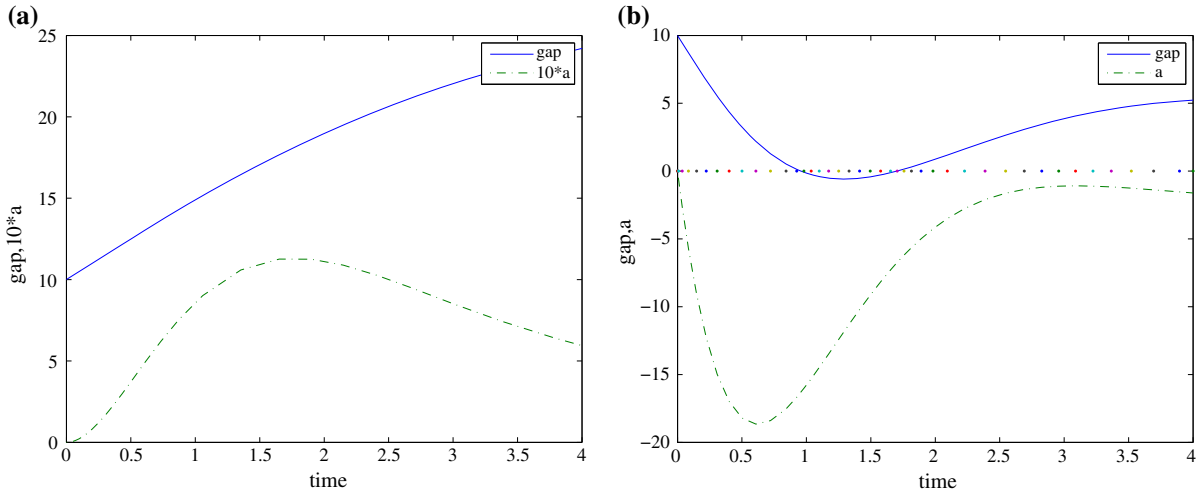


Fig. 4 Simulations of the adaptive cruise control system. **a** Plot of the gap and the scaled acceleration $10a$ of the rear car for initial conditions $v_f = 20$ and $v = 15$, which satisfies our computed safety condition. As expected, the cars do not collide. **b** Plot of the gap and the acceleration a of the rear car for initial conditions $v_f = 15$ and $v = 30$, which satisfies our computed safety condition. Surprisingly, the cars *do* collide

c_1, c_2 and the existentially quantified λ_3. Their Boolean structure is quite complicated with a maximal depth of 13. Using CAD-based techniques [5] and the assumption that both c_1 and c_2 are positive, we manage to equivalently simplify the 33 out of the 584 subformulas to the following condition, which is quantifier-free and contains only the initial velocity c_2 of the rear car:

$$\varphi'_{33} \doteq c_2^2 - 30c_2 - 75 \leq 0.$$

It is easy to see that for positive c_2 our φ'_{33} can be equivalently simplified to $\varphi''_{33} \doteq c_2 \leq 10\sqrt{3} + 15$. Since $\mathbb{R} \models \varphi''_{33} \longleftrightarrow \varphi'_{33}$, $\mathbb{R} \models \varphi'_{33} \longrightarrow \varphi'_{584}$ and $\mathbb{R} \models \varphi'_{584} \longleftrightarrow \varphi$, our condition φ''_{33} is sufficient for φ. We have established collision freedom for initial $v \leq 10\sqrt{3} + 15 \approx 32.32$ of the rear car without imposing any restriction on the dynamics of the leading car. The entire computation takes about 1 min of CPU time. Figure 4a shows a simulation of the system for admissible initial velocities $v_f = 20$ and $v = 15$.

However, there are limitations with our model. We have assumed that the system dynamics in Fig. 3 remains always satisfied. This appears natural, because it is about physical limitations, viz. maximal possible acceleration and braking, of the cars. Let us consider a scenario in which the leading car starts at velocity $v_f = 15$ decelerating at $a_f = -3$, and the rear car starts at velocity 30 with zero acceleration and an initial gap of 10. This satisfies both the system dynamics and our computed safety condition. Surprisingly, a simulation plot in Fig. 4b shows that the cars collide. This is compatible with our model, because the system dynamics is violated: The control law causes the rear car to brake harder than -5. There are several ways out. One could switch off cruise control and alert the driver when its computed acceleration values cannot be realized in reality. One could also refine our model to assume that in such cases acceleration goes exactly to the physical limits.

3.3 Life Sciences

In chemistry and systems biology there is a considerable interest in Hopf bifurcations due to their relationship with the occurrence of oscillations. Although the nature of that relationship is very subtle [3,43] there is a consensus that relevant oscillations typically occur in the presence of Hopf bifurcations, and considerable work has been done to investigate various chemical and biological systems with respect to Hopf bifurcation fixed points [24,42,44,55,67].

Numerical packages like AUTO[17] or XPPAUT[18] can locate Hopf bifurcations but cannot prove their absence. Furthermore, they might miss bifurcations that exist only for small ranges of parameters, which is not only of theoretical interest in the context of chemical and biological reaction systems. Whereas existence of Hopf bifurcations is known to be decidable [25,33,38,43] early symbolic investigations carried out for specific parameterized polynomial vector fields arising from larger examples [3,25] were not fully algorithmic but required a sequence of symbolic computations intervened with ad hoc insights and decisions made by a human, and sometimes with sophisticated coordinate transformations. Our example discussed here dates back to 2008. It is to our knowledge the first fully automatic symbolic investigation of Hopf bifurcations for a biologically relevant model [57,64].[19]

Boulier et al. have studied a model related to the gene regulatory network controlling the circadian clock of a certain unicellular green alga [3]. Figure 5 gives an overview. We are going to summarize the development of a simplified system of differential equations in [3], which will serve as a starting point for our methods discussed here. The following system describes the dynamics of the reaction system in Fig. 5a:

$$\begin{aligned}
\dot{G} &= \vartheta \cdot (\gamma_0 - G) - \alpha G P_n \\
\dot{M} &= \varrho_f G + \varrho_b \cdot (\gamma_0 - G) - \delta_M M \\
\dot{P} &= \beta M - \delta_P P + 2A_1 + A_2 + \cdots + A_{n-1} \\
\dot{P}_n &= -A_{n-1} + \vartheta \cdot (\gamma_0 - G) - \alpha G P_n \\
\dot{P}_i &= -A_{i-1} + A_i \quad \text{for} \quad 2 \le i \le n-1.
\end{aligned}$$

Lowercase greek letters are constants, and uppercase roman letters are functions in time. All constants except γ_0 correspond to the reaction rates in Fig. 5a; $G \in [0, \ldots, \gamma_0]$ is the averaged state of the gene, where $G = 0$ denotes that a polymer is bound to the gene promoter, and $G = \gamma_0$ denotes that this is not the case; M is the corresponding concentration of active messenger RNA; P, P_2, …, P_n are concentrations of the protein and its polymers; A_i is defined as $\frac{1}{\varepsilon}\left(\kappa_{i+1}^- P_{i+1} - \kappa_{i+1}^+ P_i P\right)$, where the factor $\frac{1}{\varepsilon}$ with small positive ε expresses the assumption that the polymerization reactions are fast compared to the other reactions. Note that all occurring constants and functions are strictly positive.

Using *quasi-steady-state* assumptions that $\dot{P}_2$, …, $\dot{P}_n$ are small and furthermore neglecting terms with a factor ε one approximates

$$\dot{P} = \beta M - \delta_P P + n\left(\vartheta(\gamma_0 - G) - \alpha G P_n\right) \quad \text{and} \quad P_n = \bar{\alpha} P^n, \quad \text{where} \quad \bar{\alpha} = \frac{\kappa_1^+ \cdots \kappa_{n-1}^+}{\kappa_1^- \cdots \kappa_{n-1}^-}.$$

This yields the following simpler system:

$$\begin{aligned}
\dot{G} &= \vartheta \cdot (\gamma_0 - G) - \alpha\bar{\alpha} G P^n \\
\dot{M} &= \varrho_f G + \varrho_b \cdot (\gamma_0 - G) - \delta_M M \\
\dot{P} &= n\vartheta(\gamma_0 - G) - n\alpha\bar{\alpha} G P^n + \beta M - \delta_P P.
\end{aligned}$$

At that point Boulier et al. apply various transformations, which we only sketch to the extent necessary to understand the variables occurring in their following further simplified system:

$$\begin{aligned}
\dot{G} &= \vartheta \cdot (\gamma_0 - G - G P^n) \\
\dot{M} &= \lambda G + \gamma_0 \mu - M \\
\dot{P} &= n\alpha \cdot (\gamma_0 - G - G P^n) + \delta \cdot (M - P).
\end{aligned}$$

The product $\alpha\bar{\alpha}$ has been replaced with α, P has been replaced with $\left(\frac{\vartheta}{\alpha}\right)^{\frac{1}{n}} P$, and new variables λ and μ have been introduced for terms $\varrho_f - \varrho_b$ and $\varrho_b \gamma_0$, respectively. The factor δ_M of M in the second equation has been eliminated

[17] http://indy.cs.concordia.ca/auto/.

[18] http://www.math.pitt.edu/~bard/xpp/xpp.html.

[19] More than 10 years earlier, Hong–Liska–Steinberg had applied quantifier elimination for testing stability of systems of differential equations [31].

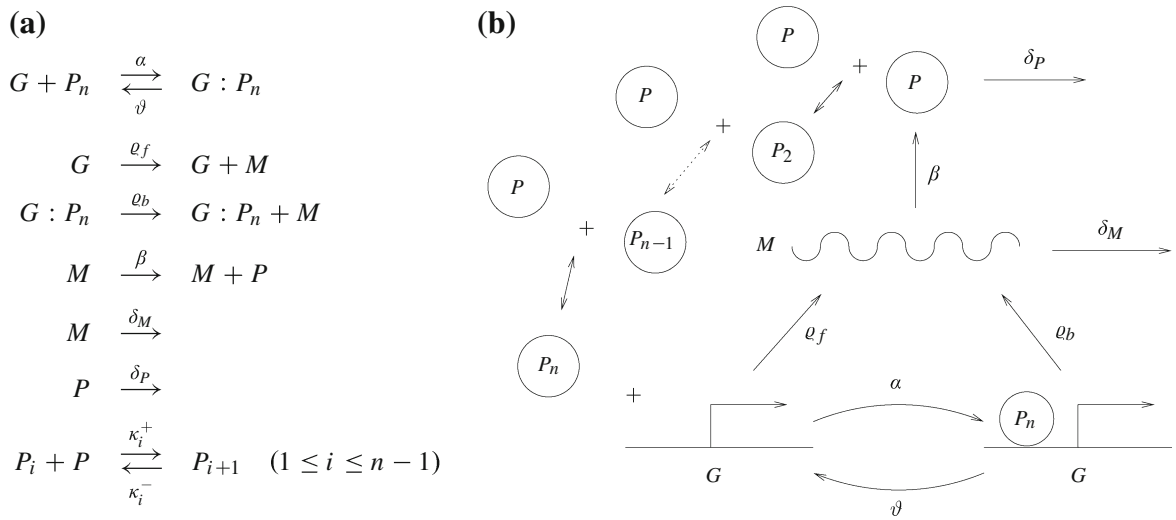


Fig. 5 A gene G regulated by a polymer obtained by combining n times a protein P: The polymer P_n binds to the gene promoter. Depending on ϱ_f and ϱ_b this can enhance or inhibit the transcription of messenger RNA M, which in turn increases or decreases the production of the protein P and its polymers, thus establishing a positive or a negative feedback loop. **a** The original reaction system. **b** A visualization of the reaction;[20] here $P_1 = P$

by rescaling time with a factor δ_M and adapting all other constants. In that course $\frac{\alpha^n}{\vartheta^{n-1}}$ has been replaced with α. Finally, M has been rescaled so that $\delta_P = \beta$, which is now called δ; again other constants had to be adapted.[21]

Notice that the new constant λ, in contrast to all other constants and functions, is not necessarily positive. In fact, for $\lambda > 0$ messenger RNA transcription is enhanced when polymer is bound to the gene promoter, while for $\lambda < 0$ it is reduced. This corresponds to the postive and negative feedback loops mentioned in Fig. 5, respectively.

From the point of view of our quantifier elimination methods that last system can be analyzed for fixed choices of $n \in \mathbb{N}$. We use a method by El Kahoui and Weber to generate from that system first-order formulas over the reals [33]. The method is applicable whenever the vector field consists of polynomials in the variables and parameters. For $n = 9$ we obtain

$$\begin{aligned}\varphi_9 \doteq\ & \exists v_1 \exists v_2 \exists v_3 \big(v_1 > 0 \wedge v_2 > 0 \wedge v_3 > 0 \wedge \vartheta > 0 \wedge \gamma_0 > 0 \wedge \mu > 0 \wedge \delta > 0 \wedge \alpha > 0 \wedge \\ & \vartheta(\gamma_0 - v_1 - v_1 v_3^9) = 0 \wedge \lambda v_1 + \gamma_0 \mu - v_2 = 0 \wedge 9\alpha(\gamma_0 - v_1 - v_1 v_3^9) + \delta(v_2 - v_3) = 0 \wedge \\ & \Delta_2 = 0 \wedge \Delta_1 > 0 \big),\end{aligned}$$

where

$$\begin{aligned}\Delta_2 \doteq\ & 162\vartheta v_3^{17}\alpha v_1 + 162\vartheta\alpha v_1 v_3^8 + 162\alpha v_1 v_3^8\delta + \vartheta + 2\vartheta v_3^9\delta + \vartheta^2 v_3^{18}\delta + \vartheta v_3^9\vartheta\delta + 81\alpha v_1 v_3^8\vartheta\delta \\ & + 81\alpha v_1 v_3^{17}\vartheta\delta + \delta^2 + \vartheta\delta^2 + \vartheta^2\delta + \vartheta^2 + 2\vartheta^2 v_3^9 + \vartheta^2 v_3^{18} + 6561\alpha^2 v_1^2 v_3^{16} + 2\vartheta^2 v_3^9\delta + \delta \\ & + 81\alpha v_1 v_3^8 + \vartheta v_3^9\delta^2 - 9\lambda\vartheta v_1 v_3^8\delta, \\ \Delta_1 \doteq\ & \vartheta\delta + \vartheta v_3^9\delta + 9\lambda\vartheta v_1 v_3^8\delta.\end{aligned}$$

We have introduced real variables v_1, v_2, and v_3 for the functions G, M, and P, respectively. The first line of our formula φ_9 collects the above-mentioned positivity conditions on the variables and parameters. The second line

[20] This image is taken from [3]. Franҫois Boulier kindly offered their source code and permitted its reproduction here.

[21] The detailed transformation can be found in [3, Section 2.1]. In a later publication Boulier et al. revised their quasi-steady state approximation to obtain another biologically more appropriate model [4]. For the applicability of our method and the quality of our results that improvement makes no difference.

(a)

n	$\exists\varphi_n$	$\exists\varphi_n[\lambda/-\lambda]$	$\exists\varphi_n[\lambda/0]$	time (s)
2	false	false	false	< 0.01
3	false	false	false	19.28
4	false	false	false	21.58
5	false	false	false	19.09
6	false	false	false	23.72
7	false	false	false	23.89
8	false	false	false	22.35
9	true	false	false	0.17
10	true	false	false	0.17

(b)

$$\begin{aligned}
\alpha &= 1\\
\delta &= 1\\
\gamma_0 &= 0.0100554964908\\
\lambda &= 17617230.5528\\
\mu &= 0\\
\vartheta &= 0.0000211443608455\\
\upsilon_1 &= 0.000000170287832189\\
\upsilon_2 &= 3\\
\upsilon_3 &= 1.24573093962
\end{aligned}$$

Fig. 6 Applying positive quantifier elimination. **a** Results and CPU times for $n = 2, \ldots, 10$ using a case distinction on the sign of λ. **b** Float approximations of exact sample solutions obtained with extended positive quantifier elimination for $n = 9$

equates the vector field of the differential equations to zero thus expressing that the system is in steady state. In the third line, Δ_2 and Δ_1 are the corresponding Hurwitz determinants of the characteristic polynomial of the Jacobian of the vector field. In general, the condition $\Delta_{n-1} = 0 \wedge \Delta_{n-2} > 0 \wedge \Delta_{n-3} > 0$ characterizes a Hopf bifurcation, and $\Delta_{n-4} > 0 \wedge \cdots \wedge \Delta_1 > 0$ in addition guarantees an empty unstable manifold; in addition, the absolute summand of the characteristic polynomial used for the Hurwitz determinants must be positive, which is not an issue with our model [33].

Hence our φ_9 is equivalent over the reals to the existence of a Hopf bifurcation with empty unstable manifold, and quantifier elimination can in principle yield necessary and sufficient conditions in terms of the parameters. In practice this turns out to be not feasible with existing implementations of real quantifier elimination. Existing CAD-based implementations cannot decompose the corresponding 9-dimensional space in reasonable time, while VS-based implementations cannot cope with the high degrees occurring here. A bit surprisingly, VS does succeed, in contrast, on the existential closures $\bar{\varphi}_n \doteq \exists\varphi_n$, although there are even more variables to eliminate. The reason is that there is more freedom with the heuristic choice of the elimination order, and powerful simplification techniques [20] help to get around solving with high degrees.

Our results for $n \in \{2, \ldots, 10\}$ are collected in a table in Fig. 6a.[22] We use extended positive quantifier elimination, which assumes that all occurring variables are strictly positive, in combination with a case distinction on the sign of λ. In the satisfiable case we obtain exact symbolic sample solutions. Figure 6b gives float approximations of such a sample solution for $\bar{\varphi}_9$.

Alternative reductions to real quantifier elimination of our model have been discussed in [71]. The vast majority of chemical and biological models as they can be found, e.g., in the BioModels database[23] is considerably larger with respect to both dimension and degrees than our example here. in Sect. 5 we are going to discuss such models based on a different translation into logic. For deciding satisfiability we will use a heuristic method, which is not based on quantifier elimination. We are going to introduce this method in the following section.

4 Subtropical Methods

We are interested in heuristically finding zeros $(z_1, \ldots, z_n) \in \mathbb{R}^n$ of large polynomials $f \in \mathbb{Z}[x_1, \ldots, x_n]$ with $z_1 > 0, \ldots, z_n > 0$. In other words, our zeros should be located strictly in the first hyper-quadrant.

Plugging in the point $(1, \ldots, 1)$ yields $f(1, \ldots, 1) = y$, with three possibilities for the sign of y. If $y = 0$, then we are done. If $y > 0$, then we consider instead $-f$, which has the same zeros as f. Therefore we may assume in

[22] For independent reasons one knows with the first Hopf bifurcation at $n = 9$ that there will be Hopf bifurcations for all $n > 9$.

[23] http://www.ebi.ac.uk/biomodels-main/.

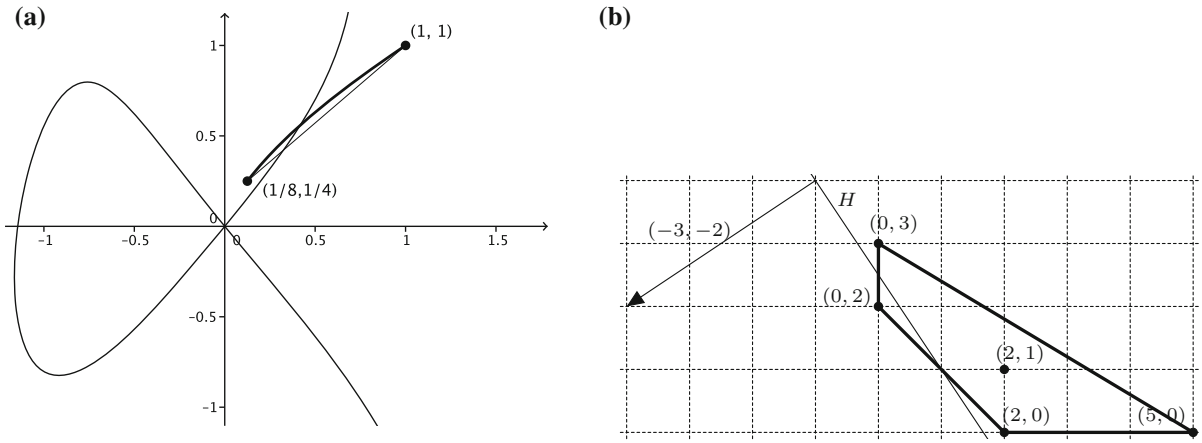


Fig. 7 Finding a zero of $f = -2x_1^5 + x_1^2x_2 - 3x_1^2 - x_2^3 + 2x_2^2$. **a** The variety of f and the segment given by $t \in [0, 2]$ of the moment curve (t^{-3}, t^{-2}) corresponding to the normal vector $(-3, -2)$ of the separating hyperplane in (**b**). **b** A subtropical view on f from (**a**). We can see a hyperplane separating $(0, 2) \in \text{newton}(f)$ from $\text{supp}(f)\backslash\{(0, 2)\}$ together with its normal vector $\mathbf{n} = (-3, -2)$

the sequel that $f(1, \ldots, 1) = y < 0$. In Sect. 4.1 we are going to find $\mathbf{p} \in \mathbb{R}^n$ with all positive coordinates such that $f(\mathbf{p}) > 0$. In Sect. 4.2 we are going to subsequently apply the intermediate value theorem to construct a zero. A detailed formal discussion of our method summarized here can be found in [63].

4.1 Heuristically Finding a Positive Point

To get a first idea consider $g = -128x_1^{10}x_2^2 + 2x_1^3x_2^9 - 4096$. For g to become positive we need a point $\mathbf{p} \in \mathbb{R}^2$ where the second term $x_1^3x_2^9$, which has a coefficient with positive sign, dominates in the sense that it has a large absolute value compared to $x_1^{10}x_2^2$ as well as compared to the coefficients -128 and -4096. This can be achieved by making x_2 larger than x_1 by a sufficient order of magnitude, like $\mathbf{p} = (2, 2^2)$ with $g(\mathbf{p}) = 2{,}093{,}056 > 0$.

We want to generalize this idea. Our discussion will be accompanied by the slightly more complicated polynomial

$$f = -2x_1^5 + x_1^2x_2 - 3x_1^2 - x_2^3 + 2x_2^2.$$

Notice that $f(1, 1) = -3 < 0$ as required by our overall framework. The slope pictured in Fig. 7a is the real variety of f, i.e., the set of all its zeros. We switch to a more abstract view of f taking into consideration only its terms and the sign of the corresponding coefficients. The terms are determined by the respective exponent vectors, and we remember the signs of the coefficients by forming a corresponding partition. Formally this gives us the *positive* and the *negative support* of f:[24]

$$\text{supp}(f) = \text{supp}^+(f) \,\dot\cup\, \text{supp}^-(f), \quad \text{supp}^+(f) = \big\{(2, 1), (0, 2)\big\}, \quad \text{supp}^-(f) = \big\{(5, 0), (2, 0), (0, 3)\big\}.$$

The support of f is pictured in Fig. 7b together with its convex hull, the *Newton polytope* of f. For our purposes here, the Newton polytope is the set of vertices of the convex hull, denoted by $\text{newton}(f) \subseteq \text{supp}(f)$. We have $(0, 2) \in \text{newton}(f) \cap \text{supp}^+(f)$. Since that point is in the convex hull there exists a separating hyperplane H with a normal vector $\mathbf{n} = (-3, -2)$ pointing outwards. In the real world the coordinates of $\mathbf{n}$ give a suitable ratio such that for (t^{-3}, t^{-2}) with sufficiently large t the term x_2^2 corresponding to $(0, 2)$ dominates in the sense of the introductory example. Plugging in increasing powers of 2 for t we find that already for $\mathbf{p} = (2^{-3}, 2^{-2}) = \left(\frac{1}{8}, \frac{1}{4}\right)$ we obtain $f(\mathbf{p}) = \frac{1087}{16{,}384} > 0$. In Fig. 7a we see the segment $\big\{(t^{-3}, t^{-2}) \in \mathbb{R}^2 \mid t \in [1, 2]\big\}$ of the corresponding moment curve, which converges to the origin as t goes to infinity.

[24] In similar contexts the support has also been called the *frame* of f.

The separating hyperplane H and its normal vector $\mathbf{n}$ can be efficiently found using linear programming techniques; see [63] for details. The method is incomplete. It fails when the positive support is non-empty but the Newton polytope contains exclusively points from the negative support; formally, $\operatorname{supp}^+(f) \neq \emptyset$ but $\operatorname{newton}(f) \subseteq \operatorname{supp}^-(f)$ and thus $\operatorname{supp}^+(f) \cap \operatorname{newton}(f) = \emptyset$. If, in contrast, $\operatorname{supp}^+(f) = \emptyset$, then f is negative definite on the first hyper-quadrant, and we can be certain that there is no zero with positive coordinates.

4.2 Constructing a Zero

We have found $f(1,1) < 0 < f\left(\frac{1}{8}, \frac{1}{4}\right)$. By the intermediate value theorem there is at least one zero on the connecting line segment from $(1,1)$ to $\left(\frac{1}{8}, \frac{1}{4}\right)$. We straightforwardly write down that (x_1, x_2) is a zero of f lying on that line segment:

$$
\begin{aligned}
&-2x_1^5 + x_1^2 x_2 - 3x_1^2 - x_2^3 + 2x_2^2 = 0 \\
&x_1 = \tfrac{1}{8} + y \cdot \left(1 - \tfrac{1}{8}\right) \\
&x_2 = \tfrac{1}{4} + y \cdot \left(1 - \tfrac{1}{4}\right), \qquad y \in \,]0, 1[.
\end{aligned}
$$

Plugging the equations for x_1 and x_2 into the first equation we obtain a univariate equation in y, which has at least one solution for $y \in \,]0, 1[$:

$$
\begin{aligned}
\bar{f} &= f\left(\tfrac{1}{8} + y \cdot \left(1 - \tfrac{1}{8}\right), \tfrac{1}{4} + y \cdot \left(1 - \tfrac{1}{4}\right)\right) \\
&= \tfrac{1}{d} \cdot \left(-16807y^5 - 12005y^4 - 934y^3 - 20778y^2 + 285y + 1087\right), \quad \text{where} \quad d \in \mathbb{N}.
\end{aligned}
$$

Univariate real root isolation yields $y \in \,]0.2, 0.3[$ and a corresponding real algebraic number $\langle \bar{f}, \,]0.2, 0.3[\rangle$ for y. Back-substitution finally yields an exact solution for our original problem $f(x_1, x_2) = 0$:

$$
\begin{aligned}
x_1 &= \langle 686x^5 - 78x^3 + 584x^2 - 150x - 13, \,]0.32, 0.33[\rangle \\
x_2 &= \langle 16807x^5 - 12005x^4 + 2026x^3 + 9122x^2 - 4609x + 323, \,]0.42, 0.43[\rangle.
\end{aligned}
$$

Figure 7a shows the connecting line segment and its intersection with the variety near $(0.32, 0.42)$. For practical purposes the isolating intervals of the real algebraic numbers can be efficiently refined to arbitrary precision.

5 Further Applications in Chemistry and the Life Sciences

In Sect. 3.3 we have studied Hopf bifurcations for a biological reaction system using results on Hurwitz determinants. The variables in our real first-order model corresponded to concentrations of species and reaction rates. Using ideas from stoichiometric network analysis [10], it is possible to analyze the system dynamics in flux space instead of concentration space and to represent the space of steady states with a combination of subnetworks. The subnetworks form a convex cone in flux space [68]. There are techniques for decomposing that cone and limiting the search for Hopf bifurcations to lower dimensional facets [22]. For our purposes here it is sufficient to understand that we are going to obtain a whole family of formulas for a single model corresponding to those facets. Methods for detecting Hopf bifurcations using similar approaches have been used in several hand computations in a semi-algorithmic way for parametric systems, the most elaborate of which is described in [25].

The formulation of a reaction system using *convex coordinates*[25] in flux space implicitly limits that system to steady states. Technically, the obtained first-order formulas need not equate the vector field to zero anymore (as in the second line the equation of φ_9 in Sect. 3.3). Instead they exclusively contain positivity conditions on all variables and conditions $\Delta_{n-1} = 0$, $\Delta_{n-2} > 0$, …, $\Delta_1 > 0$ on Hurwitz determinants. Again, all variables are existentially

[25] In some of the cited publications convex coordinates are called *reaction coordinates*.

quantified. Generally Δ_{n-1} is considerably larger than $\Delta_{n-2}, \ldots, \Delta_1$. Recall that for a Hopf bifurcation, besides the equation $\Delta_{n-1} = 0$, only the two inequalities $\Delta_{n-2} > 0$ and $\Delta_{n-3} > 0$ are relevant, while the other inequalities guarantee an empty unstable manifold.

We are going to apply the subtropical method described in the previous section for finding zeros of Δ_{n-1} with all positive coordinates, and then checking for those zeros the validity of the inequalities $\Delta_{n-2} > 0, \ldots, \Delta_1 > 0$. For details on the modeling and also on the reaction systems discussed below see [21–23].

5.1 Methylene Blue Oscillator (MBO)

Our discussion of the autocatalytic system Methylene Blue Oscillator (MBO) follows [21]. An alternative formulation using the same approach can be found in [23]. The original MBO reaction system comprises 15 reactions and 11 species, not counting water: O_2, O_2^-, HS, MB^+, MB, MBH, HS^-, OH^-, S, H_2O_2, and HO_2^-. Figure 8 shows that original system in combination with an application of methylene blue as a dye.

The system can be reduced to a 6-dimensional system considering only the essential species O_2, O_2^-, HS, MB^+, MB, and MBH. This yields us a collection of 496 polynomial equations $\Delta_{n-1} = 0$ corresponding to facets of various dimensions.[26] A typical polynomial Δ_{n-1} in that collection has 7 variables with degrees between 4 and 9 and around 6000 summands. The polynomials are essentially irreducible.[27] There is a Hopf bifurcation if at least one of those equations has a solution with all positive coordinates that also satisfies the corresponding conditions $\Delta_{n-2} > 0$ and $\Delta_{n-3} > 0$. Using the subtropical approach from Sect. 4 we discovered a suitable zero in 144 cases (30%), where the inequalities were always satisfied. We could exclude the existence of a zero in 338 cases (67%). The method failed due to its incompleteness in only 14 cases (3%). The overall computation time was 200 s.

(a)

$$
\begin{aligned}
MB^+ + HS^- &\longrightarrow MB + HS \\
H_2O + MB + HS^- &\longrightarrow MBH + HS + OH^- \\
HS + OH^- + MB^+ &\longrightarrow MB + S + H_2O \\
H_2O + 2MB &\longrightarrow MB^+ + MBH + OH^- \\
HS^- + O_2 &\longrightarrow HS + O_2^- \\
HS + O_2 + OH^- &\longrightarrow O_2^- + S + H_2O \\
H_2O + HS^- + O_2^- &\longrightarrow HO_2^- + HS + OH^- \\
O_2^- + HS &\longrightarrow HO_2^- + S \\
H_2O_2 + 2HS^- &\longrightarrow 2HS + 2OH^- \\
MB + O_2 &\longrightarrow MB^+ + O_2^- \\
HS^- + MB + H_2O_2 &\longrightarrow MB^+ + HS + 2OH^- \\
OH^- + 2HS &\longrightarrow HS^- + S + H_2O \\
MB + HS &\longrightarrow MBH + S \\
H_2O + MBH + O_2^- &\longrightarrow MB + H_2O_2 + OH^- \\
&\longrightarrow O_2
\end{aligned}
$$

(b)

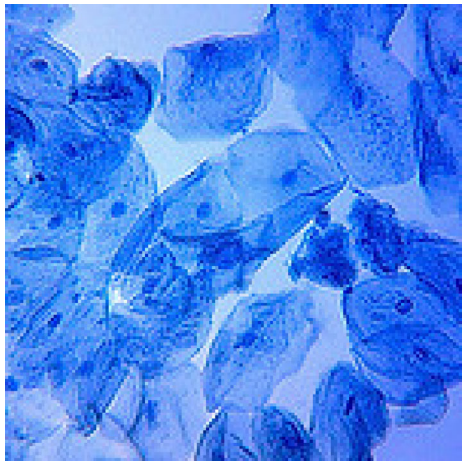


Fig. 8 The methylene blue oscillator (MBO). **a** The original chemical reaction system. **b** Human cheek cells dyed with methylene blue[28]

5.2 Mitogen-Activated Protein Kinase (MAPK)

The largest reaction system considered with our approach so far is the Mitogen-Activated Protein Kinase (MAPK) cascade, which is well studied in cell biology. Details can be found in [23].

[26] Available at http://research-data.redlog.eu/ISSAC2015/OMBO/.

[27] In some cases a single variable can be factored off.

[28] *Human Cheek Cells* by Joseph Elsbernd is licensed under CC-BY 2.0 at https://www.flickr.com/photos/codonaug/6936088946/in/album-72157629826384173/.

Here we obtain a collection of 21 polynomial equations of the form $\Delta_{n-1} = 0$.[29] The largest polynomial $\Delta_{n-1} = 0$ in that collection has 10 variables with degrees between 5 and 12 and around 863,000 summands.[30] We discovered a suitable zero in 5 cases, including that largest polynomial, and we could exclude the existence of a zero in all remaining 16 cases.

The overall computation time was 16 s, 15 s of which were spent on the above-mentioned largest polynomial, where we found a negative and a positive point in 10 s and computed the zero in another 5 s. To give an impression of the speed of our methods, we mention that reading and parsing that polynomial from a file takes 25 s, and, e.g., determining the degrees of its 10 variables would take 2 s.

In the positive case, our subtropical approach is capable of finding many suitable zeros in a model: On the one hand, $\mathrm{supp}^+(f) \cap \mathrm{newton}(f)$ typically contains more than one point. On the other hand, with each point in $\mathrm{supp}^+(f) \cap \mathrm{newton}(f)$ we can follow the corresponding moment curve to obtain infinitely many solutions. Unfortunately, with MAPK we did not succeed in finding any zero that also satisfies the necessary Hopf conditions $\Delta_{n-2} > 0$ and $\Delta_{n-3} > 0$ so far.

6 Concluding Remarks

The quantifier elimination methods used throughout this article offer the expressivity and flexibility of first-order logic. On the other hand, these methods are deeply rooted in symbolic computation, specifically in real algebra. This allows to make use of the rich algorithmic infrastructure coming with state-of-the-art computer algebra systems. We have seen applications of our methods ranging from geometry via verification to the natural sciences. It is noteworthy that with all our applications we never used regular real quantifier elimination but always certain variants. However, those variants have not been designed for our particular applications. Instead they belong to a collection of established tools, applicable to many more domains than we have discussed here. As a counterpart to the very general concept of quantifier elimination we have seen quite specialized subtropical methods. The development of those methods was triggered by the success of quantifier elimination-based methods in chemistry and in the life sciences. We believe that, similarly to real quantifier elimination and its variants, subtropical methods have the potential to develop into a way more general tool than it might appear now. An important next step is the subtropical treatment of input with one or several inequalities as side conditions in both theory and software.

Acknowledgements Open access funding provided by Max Planck Society.

References

1. Arnon, D.S.: Algorithms for the geometry of semi-algebraic sets. Technical Report 436, Computer Science Department, University of Wisconsin-Madison, Ph.D. Thesis (1981)
2. Basu, S., Pollack, R., Roy, M.-F.: On the combinatorial and algebraic complexity of quantifier elimination. J. ACM **43**(6), 1002–1045 (1996)
3. Boulier, F., Lefranc, M., Lemaire, F., Morant, P.-E., Ürgüplü, A.: On proving the absence of oscillations in models of genetic circuits. In: Proceedings of the AB 2007, volume 4545 of LNCS, pp. 66–80. Springer (2007)
4. Boulier, F., Lefranc, M., Lemaire, F., Morant, P.-E.: Applying a rigorous quasi-steady state approximation method for proving the absence of oscillations in models of genetic circuits. In: Proceedings of the AB 2008, volume 5147 of LNCS, pp. 56–64. Springer (2008)

[29] Available at http://research-data.redlog.eu/ISSAC2015/MAPK/.

[30] Typesetting that polynomial using the standard LaTeX article class with 10 pt font size and A4 paper requires more than 3000 pages.

5. Brown, C.W., Gross C.: Efficient preprocessing methods for quantifier elimination. In: Proceedings of the CASC 2006, volume 4194 of LNCS, pp. 89–100. Springer (2006)
6. Brown, C.W.: QEPCAD B: a program for computing with semi-algebraic sets using CADs. ACM SIGSAM Bull. **37**(4), 97–108 (2003)
7. Brown, C.W., Košta, M.: Constructing a single cell in cylindrical algebraic decomposition. J. Symb. Comput. **70**, 14–48 (2014)
8. Buchberger, B.: Ein Algorithmus zum Auffinden der Basiselemente des Restklassenringes nach einem nulldimensionalen Polynomideal. Doctoral dissertation, Mathematical Institute, University of Innsbruck, Innsbruck, Austria (1965)
9. Chou, S.-C.: Mechanical Geometry Theorem Proving. Mathematics and Its Applications. D. Reidel Publishing Company, Dordrecht, Boston, Lancaster, Tokyo (1988)
10. Clarke, B.L.: Stability of complex reaction networks. In: Prigogine, I., Rice, Stuart A. (eds.) Advances in Chemical Physics, vol. 43. Wiley, Hoboken (1980)
11. Collins, G.E.: Quantifier elimination for real closed fields by cylindrical algebraic decomposition—preliminary report. ACM SIGSAM Bull. **8**(3), 80–90 (1974). Proc. EUROSAM '74
12. Collins, G.E.: Quantifier elimination for the elementary theory of real closed fields by cylindrical algebraic decomposition. In: Automata Theory and Formal Languages. 2nd GI Conference, volume 33 of LNCS, pp. 134–183. Springer (1975)
13. Collins, G.E.: Quantifier elimination by cylindrical algebraic decomposition—twenty years of progress. In: Caviness, B.F., Johnson, J.R. (eds.) Quantifier Elimination and Cylindrical Algebraic Decomposition, pp. 8–23. Springer, Berlin (1998)
14. Collins, G.E., Hong, H.: Partial cylindrical algebraic decomposition for quantifier elimination. J. Symb. Comput. **12**(3), 299–328 (1991)
15. Davenport, J.H., Heintz, J.: Real quantifier elimination is doubly exponential. J. Symb. Comput. **5**(1–2), 29–35 (1988)
16. Davis, M.: Mathematical Procedures for Decision Problems. Final Report on Ordnance Research and Development Project No. TB2-0001 (1954)
17. Dolzmann A., Sturm T. Redlog User Manual, 2nd edn. Technical Report MIP-9905, FMI, Universität Passau, Germany (1999)
18. Dolzmann, A., Sturm, T., Weispfenning, V.: A new approach for automatic theorem proving in real geometry. J. Autom. Reason. **21**(3), 357–380 (1998)
19. Dolzmann, A., Sturm, T.: Redlog: computer algebra meets computer logic. ACM SIGSAM Bull. **31**(2), 2–9 (1997)
20. Dolzmann, A., Sturm, T.: Simplification of quantifier-free formulae over ordered fields. J. Symb. Comput. **24**(2), 209–231 (1997)
21. Errami, H., Eiswirth, M., Grigoriev, D., Seiler, W.M., Sturm, T., Weber, A.: Efficient methods to compute Hopf bifurcations in chemical reaction networks using reaction coordinates. In: Proceedings of the CASC 2013, volume 8136 of LNCS, pp. 88–99. Springer (2013)
22. Errami, H., Seiler, W.M., Eiswirth, M., Weber, A.: Computing Hopf bifurcations in chemical reaction networks using reaction coordinates. In: Proceedings of the CASC 2012, volume 7442 of LNCS. Springer (2012)
23. Errami, H., Eiswirth, M., Grigoriev, D., Seiler, W.M., Sturm, T., Weber, A.: Detection of Hopf bifurcations in chemical reaction networks using convex coordinates. J. Comput. Phys. **291**, 279–302 (2015)
24. Fussmann, G.F., Ellner, S.P., Shertzer, K.W., Hairston Jr., N.G.: Crossing the Hopf bifurcation in a live predator–prey system. Science **290**(5495), 1358–1360 (2000)
25. Gatermann, K., Eiswirth, M., Sensse, A.: Toric ideals and graph theory to analyze Hopf bifurcations in mass action systems. J. Symb. Comput. **40**(6), 1361–1382 (2005)
26. Godbole, D.N., Lygeros, J.: Longitudinal control of the lead car of a platoon. IEEE Trans. Veh. Technol. **43**(4), 1125–1135 (1994)
27. Grigoriev, D.: Complexity of deciding Tarski algebra. J. Symb. Comput. **5**(1–2), 65–108 (1988)
28. Gulwani, S., Tiwari, A.: Constraint-based approach for analysis of hybrid systems. In: Proceedings of the CAV 2008, volume 5123 of LNCS, pp. 190–203. Springer (2008)
29. Hilbert, D.: Grundlagen der Geometrie, 13th edn. Teubner Studienbücher Mathematik. Teubner, Stuttgart (1987)
30. Hong, H.: Comparison of several decision algorithms for the existential theory of the reals. Technical Report 91-41.0, RISC, Johannes Kepler University, A-4040 Linz, Austria (1991)
31. Hong, H., Liska, R., Steinberg, S.: Testing stability by quantifier elimination. J. Symb. Comput. **24**(2), 161–187 (1997)
32. Jirstrand, M.: Cylindrical algebraic decomposition—an introduction. Technical Report 1995-10-18, Department of Electrical Engineering, Linköping University, Linköping, Sweden (1995)
33. Kahoui, M.El, Weber, A.: Deciding Hopf bifurcations by quantifier elimination in a software-component architecture. J. Symb. Comput. **30**(2), 161–179 (2000)
34. Kapur, D.: Using Gröbner bases to reason about geometry problems. J. Symb. Comput. **2**(4), 399–408 (1986)
35. Košta, M.: New concepts for real quantifier elimination by virtual substitution. Doctoral dissertation, Saarland University, Germany (2016)
36. Košta, M., Sturm, T., Dolzmann, A.: Better answers to real questions. J. Symb. Comput. **74**, 255–275 (2016)
37. Kutzler, B.A., Stifter, S.: On the application of Buchberger's algorithm to automated geometry theorem proving. J. Symb. Comput. **2**(4), 389–397 (1986)
38. Liu, W.-M.: Criterion of Hopf bifurcations without using eigenvalues. J. Math. Anal. Appl. **182**(1), 250–256 (1994)
39. Loos, R., Weispfenning, V.: Applying linear quantifier elimination. Comput. J. **36**(5), 450–462 (1993)
40. McCallum, S.: An improved projection operation for cylindrical algebraic decomposition of three-dimensional space. J. Symb. Comput. **5**(1–2), 141–161 (1988)

41. McPhee, N.F., Chou, S.-C., Gao, X.-S.: Mechanically proving geometry theorems using a combination of Wu's method and Collins' method. In: Proceedings of CADE-12, volume 814 of LNAI, pp. 401–415. Springer (1994)
42. Mincheva, M., Roussel, M.R.: Graph-theoretic methods for the analysis of chemical and biochemical networks. I. Multistability and oscillations in ordinary differential equation models. J. Math. Biol. **55**(1), 61–86 (2007)
43. Niu, W., Wang, D.: Algebraic approaches to stability analysis of biological systems. Math. Comput. Sci. **1**(3), 507–539 (2008)
44. Novak, B., Pataki, Z., Ciliberto, A., Tyson, J.J.: Mathematical model of the cell division cycle of fission yeast. Chaos **11**(1), 277–286 (2001)
45. Prajna, S., Jadbabaie, A., Pappas, G.J.: A framework for worst-case and stochastic safety verification using barrier certificates. IEEE Trans. Autom. Control **52**(8), 1415–1428 (2007)
46. Prestel, A.: Lectures on formally real fields, volume 1093 of Lecture Notes in Mathematics. Springer (1984)
47. Puri, A., Varaiya, P.: Driving safely in smart cars. In: Proceedings of the 1995 American Control Conference. IEEE (1995)
48. Renegar, J.: On the computational complexity and geometry of the first-order theory of the reals. Part II: the general decision problem. Preliminaries for quantifier elimination. J. Symb. Comput. **13**(3), 301–328 (1992)
49. Ritt, J.F.: Differential Equations from the Algebraic Standpoint, volume 14 of Colloquium Publications. American Mathematical Society, New York (1932)
50. Ritt, J.F.: Differential Algebra, volume 33 of Colloquium Publications. American Mathematical Society, Providence (1950)
51. Seidenberg, A.: An elimination theory for differential algebra. Univ. Calif. Publ. Math. New Ser. **3**(2), 31–66 (1956)
52. Seidenberg, A.: Some remarks on Hilbert's Nullstellensatz. Arch. Math. **7**(4), 235–240 (1956)
53. Seidenberg, A.: On k-constructable sets, k-elementary formulae, and elimination theory. J. für die reine und angewandte Math. **239–240**, 256–267 (1969)
54. Seidl, A., Sturm, T.: A generic projection operator for partial cylindrical algebraic decomposition. In: Proceedings of the ISSAC 2003, pp. 240–247. ACM (2003)
55. Sensse, A., Hauser, M.J.B., Eiswirth, M.: Feedback loops for Shilnikov chaos the peroxidase–oxidase reaction. J. Chem. Phys. **125**(1), 014901-1–014901-12 (2006)
56. Sturm, T., Tiwari, A.: Verification and synthesis using real quantifier elimination. In: Proceedings of the ISSAC 2011, pp. 329–336. ACM (2011)
57. Sturm, T., Weber, A.: Investigating generic methods to solve Hopf bifurcation problems in algebraic biology. In: Proceedings of the AB 2008, volume 5147 of LNCS, pp. 200–215. Springer (2008)
58. Sturm, T., Weispfenning, V.: Computational geometry problems in Redlog. In: Automated Deduction in Geometry, volume 1360 of LNAI, pp. 58–86. Springer (1998)
59. Sturm, T., Weispfenning, V.: Rounding and blending of solids by a real elimination method. In: Proceedings of the IMACS World Congress 1997, volume 2, pp. 727–732. Wissenschaft & Technik Verlag, Berlin (1997)
60. Sturm, T.: An algebraic approach to offsetting and blending of solids. In: Proceedings of the CASC 2000, pp. 367–382. Springer (2000)
61. Sturm, T.: New domains for applied quantifier elimination. In: Proceedings of the CASC 2006, volume 4194 of LNCS. Springer (2006)
62. Sturm, T.: Real Quantifier Elimination in Geometry. Doctoral dissertation, Universität Passau, Germany (1999)
63. Sturm, T.: Subtropical real root finding. In: Proceedings of the ISSAC 2015, pp. 347–354. ACM (2015)
64. Sturm, T., Weber, A., Abdel-Rahman, E.O., El Kahoui, M.: Investigating algebraic and logical algorithms to solve Hopf bifurcation problems in algebraic biology. Math. Comput. Sci. **2**(3), 493–515 (2009)
65. Tarski, A.: A decision method for elementary algebra and geometry. Prepared for publication by J. C. C. McKinsey. In: RAND Report R109, August 1948, Revised May 1951, 2nd Edition, RAND (1957)
66. Tiwari, A.: Approximate reachability for linear systems. In: Proceedings of the HSCC 2003, volume 2623 of LNCS, pp. 514–525. Springer (2003)
67. Tyson, J.J., Chen, K., Novak, B.: Network dynamics and cell physiology. Nat. Rev. Mol. Cell Biol. **2**(12), 908–916 (2001)
68. Wagner, C., Urbanczik, R.: The geometry of the flux cone of a metabolic network. Biophys. J. **89**(6), 3837–3845 (2005)
69. Wang, D.: Reasoning about geometric problems using an elimination method. In: Automated Practical Reasoning, Texts and Monographs in Symbolic Computation, pp. 147–185. Springer (1995)
70. Wang, D.: An elimination method for polynomial systems. J. Symb. Comput. **16**(2), 83–114 (1993)
71. Weber, A., Sturm, T., Abdel-Rahman, E.O.: Algorithmic global criteria for excluding oscillations. Bull. Math. Biol. **73**(4), 899–916 (2011)
72. Weispfenning, V.: The complexity of linear problems in fields. J. Symb. Comput. **5**(1–2), 3–27 (1988)
73. Weispfenning, V.: Quantifier elimination for real algebra—the quadratic case and beyond. Appl. Algebra Eng. Commun. Comput. **8**(2), 85–101 (1997)
74. Wu, W.-T.: Basic principles of mechanical theorem proving in elementary geometries. J. Syst. Sci. Math. Sci. **4**(3), 207–235 (1984)
75. Wu, W.-T.: Basic principles of mechanical theorem proving in elementary geometries. J. Autom. Reason. **2**(3), 219–252 (1986)